


CERTIFICATE

I hereby certify that the attached patent application (along with any other paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on this date December 7, 2001, in an envelope as "Express Mail Post Office to Addressee" Mailing Label Number EL798606343US addressed to the: Box Patent Application, Assistant Commissioner for Patents, Washington, D.C. 20231.

Date: 12-7-01
Himanshu S. Amin**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re patent application of:

Applicant: Jason L. Zander

Examiner: D. Jung

Serial No: Continuation of 09/178,907

Art Unit: 2171

Filing Date: October 26, 1998

Title: INSTALLABLE SCHEMA FOR LOW-OVERHEAD DATABASES

Box Patent Application
Assistant Commissioner for Patents
U.S. Patent and Trademark Office
Washington, D.C. 20231

PRELIMINARY AMENDMENT

Dear Sir:

Prior to performing substantive examination of the subject continuation application, entry of the herein amendments is respectfully requested.

CLEAN VERSION OF REPLACEMENT PARAGRAPHS TO THE
SPECIFICATION AND ALL PENDING CLAIMS

In the Specification:

Please insert the following statement at line 3, page 1 of the specification:

Cross Reference to Related Application

The present application is a continuation application of U.S. Patent Application Ser. No. 09/178,907, filed October 26, 1998, and entitled INSTALLABLE SCHEMA FOR LOW-OVERHEAD DATABASES.

Please replace the paragraph beginning on page 6, line 15, with the following rewritten paragraph:

The first line of file 210 declares the schema definition. The subsequent lines define the tables, the columns of each table, and the data type of each column in a conventional manner.

```
declare schema <Name>, <version>, <sid>;
```

declares a schema by specifying a mnemonic name, a version number, and a schema identifier, which is a conventional 16-byte globally unique identifier (GUID). The above statement is the only new syntax required by the invention; the remainder closely follows standard ANSI SQL syntax.

Please replace the paragraph beginning on page 7, line 16, with the following rewritten paragraph:

A compiler 220 translates definition script file 210 into three different kinds of files 230 representing the schema. A standalone binary (.clb) schema file 231 contains a compiled version of the schema data that can be referenced directly. This file is needed only to provide schema definitions and data to applications that did not have access to file 232, below, when that file was compiled. It can be installed on a user's computer separately from any other program, in a location where the database engine can find it.

Please replace the paragraph beginning on page 9, line 4, with the following rewritten paragraph:

Where numerous different databases 310 reside on the same computer, a generic browser program 320 allows a user to invoke database engine 330 to query and modify data on any of the databases. Although it can access any schema installed in the database itself, a system according to the invention need not have an internal schema. To provide for this case, case (c) above, database 310 includes a pointer 311 to standalone schema file 231 for that database. The first 16 bytes of the pointer contain a standard globally unique identifier (GUID) for the particular schema file 231. This requires that the schema files 231 for all databases 310 on the computer be accessible to the computer; they can be stored in a local catalog if desired. Although the generic browsing capability requires storing a small pointer 311 to the schema in the database, it avoids the overhead of storing the very much larger schema itself in the database. In some cases, it might be desired to make a database opaque to all but authorized browsers; merely eliminating the file 231 for such a database then prohibits generic browsers from accessing the data.

In the Claims:

Please cancel claims 15-39.

Please add new claims 40-52 as indicated below.

The currently pending claims read as follows:

1. A method for processing relational databases having data files, comprising:
defining a schema of the databases as a script;
compiling the script file into a representation of the schema;
constructing an executable application program for processing the databases; and
installing the representation of the schema in the executable application program such that the schema is stored separately from the data file of the relational databases processed by the application program.

2. A medium containing instructions and data for executing the method of claim 1 on a programmable digital computer.
3. A method according to claim 1, wherein the schema includes definitions of tables, columns for the tables, and data types for the columns.
4. A method according to claim 1, wherein the schema further includes at least one index.
5. A method according to claim 3, further comprising compiling the schema script into a helper file containing structural definitions for the tables and a set of macros.
6. A method according to claim 1, further comprising compiling the script file into a further standalone representation of the schema independent of the first representation.
7. A method according to claim 6, further comprising installing the standalone representation in a user computer independently of the application program.
8. A method for processing relational databases with an application program, comprising:
recording a desired schema for the databases as a human readable script in a source format;
and
compiling the script file into a representation capable of being included in the code of the application program, such that the schema file forms a part of the application program rather than a part of the databases.
9. A medium containing instructions and data for executing the method of claim 8 on a programmable digital computer.
10. A method according to claim 8, wherein the representation includes a globally unique schema identifier.

-
11. A method according to claim 10, wherein the representation further includes a version number.
12. A method according to claim 8, wherein the schema includes definitions of tables and columns for the tables.
13. A method according to claim 12, wherein the schema further includes at least one index.
14. A method according to claim 12, wherein the representation includes structural constructs representing the definitions of the tables and columns.
40. (New) A method according to claim 1, the script being a single stand-alone script file.
41. (New) A method for processing relational databases having data files organized according to a defined schema, comprising:
defining application program code for performing at least one function upon the databases;
and
including with the application code a schema file separate from the data files and representing the schema of the data files, such that the function operates upon the data files in accordance with the schema wherein the application code accesses a further program for executing the function, and wherein the application code passes a pointer to the schema file to the further code for locating particular data in the data files.
42. (New) A medium containing instructions and data for executing the method of claim 41 on a programmable digital computer.
43. (New) A method according to claim 41, wherein the further program is a database engine.
44. (New) A method according to claim 41, wherein the schema file includes structural

constructs representing definitions of the tables and columns in the schema.

45. (New) A method according to claim 44, wherein the schema file further includes at least one index.

46. (New) A method according to claim 45, wherein the constructs are separately integrated into the application code.

47. (New) A system for processing a relational database, comprising:
a data file for containing data for the database according to a schema;
a schema defined as a single stand-alone script file that provides a schema definition for the database; and
a compiler responsive to the single stand-alone script file for producing a representation of the schema independent from the data file.

48. (New) A system according to claim 47, wherein the representation is a binary representation.

49. (New) A system according to claim 48, wherein the schema definition includes specifications of tables and columns.

50. (New) A system according to claim 49, wherein the representation includes constructs embodying the specifications of the tables and columns.

51. (New) A system for processing a relational database, comprising:
a data file containing relational data organized according to a schema;
an application program separate from the data file and including a representation of the schema of the data file, code responsive to the representation for performing an operation on the data file in accordance with the schema; and
a further program that operates upon the data file in accordance with the schema, the

application program passing a pointer to the schema file to the further program for accessing data in the data file.

52. (New) A system according to claim 51, the further program comprising a database engine that executes the operation upon the data file in response to receiving the pointer to the schema from the application program.

REMARKS

Claims 1-14 and 40-52 are currently pending and under consideration. Claims 15-39 have been cancelled. Claims 40-52 have been newly added. A clean version of replacement paragraphs and all pending claims are found at pages 2-7. The specification has been amended herein. A marked up version of amended paragraphs are found at pages 9-11.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063.

The Examiner is invited to contact applicants' undersigned representative over the telephone to expedite favorable prosecution of the subject application.

Respectfully submitted,
AMIN & TUROCY, LLP



Himanshu S. Amin
Reg. No. 40,894

AMIN & TUROCY, LLP
24th Floor, National City Center
1900 East 9th Street
Cleveland, Ohio 44114
Telephone: (216) 696-8730
Facsimile: (216) 696-8731

VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the Specification:

Please amend the specification as follows:

Please insert the following statement at line 3, page 1 of the specification:

-- Cross Reference to Related Application

The present application is a continuation application of U.S. Patent Application Ser. No. 09/178,907, filed October 26, 1998, and entitled INSTALLABLE SCHEMA FOR LOW-OVERHEAD DATABASES.--

Page 6, line 17, delete "[JLZ1]".

Page 7, line 20, delete first two occurrences of "It".

Page 9, line 8, replace "fist" with -- first --.

In the Claims:

Please cancel claims 15-39.

Please add new claims 40-52 as indicated below.

--40. (New) A method according to claim 1, the script being a single stand-alone script file.

41. (New) A method for processing relational databases having data files organized according to a defined schema, comprising:

defining application program code for performing at least one function upon the databases;

and

including with the application code a schema file separate from the data files and representing the schema of the data files, such that the function operates upon the data files in accordance with the schema wherein the application code accesses a further program for executing the function, and

wherein the application code passes a pointer to the schema file to the further code for locating particular data in the data files.

42. (New) A medium containing instructions and data for executing the method of claim 41 on a programmable digital computer.

43. (New) A method according to claim 41, wherein the further program is a database engine.

44. (New) A method according to claim 41, wherein the schema file includes structural constructs representing definitions of the tables and columns in the schema.

45. (New) A method according to claim 44, wherein the schema file further includes at least one index.

46. (New) A method according to claim 45, wherein the constructs are separately integrated into the application code.

47. (New) A system for processing a relational database, comprising:
a data file for containing data for the database according to a schema;
a schema defined as a single stand-alone script file that provides a schema definition for the database; and
a compiler responsive to the single stand-alone script file for producing a representation of the schema independent from the data file.

48. (New) A system according to claim 47, wherein the representation is a binary representation.

49. (New) A system according to claim 48, wherein the schema definition includes specifications of tables and columns.

50. (New) A system according to claim 49, wherein the representation includes constructs embodying the specifications of the tables and columns.

51. (New) A system for processing a relational database, comprising:
a data file containing relational data organized according to a schema;
an application program separate from the data file and including a representation of the schema of the data file, code responsive to the representation for performing an operation on the data file in accordance with the schema; and
a further program that operates upon the data file in accordance with the schema, the application program passing a pointer to the schema file to the further program for accessing data in the data file.

52. (New) A system according to claim 51, the further program comprising a database engine that executes the operation upon the data file in response to receiving the pointer to the schema from the application program.--